CLAIMS

What is claimed is:

1. A method for processing changes to orders in an order processing system, the method comprising the steps of:

receiving a change to an existing order;

generating a change order based on the existing order, the change order containing the change to the existing order;

comparing the change order to the existing order to generate a change order result that indicates differences between the change order and the existing order; and

providing the change order result to at least one recipient such that the recipient may distinguish the differences between the change order and the existing order.

2. The method of claim 1 wherein the step of generating a change order containing the change to the existing order comprises the steps of:

copying the existing order to the change order such that the change order contains any objects that exist within the existing order, each object having at least one attribute and an associated value; and

replacing values of any attributes of objects in the change order with new values for those attributes as indicated in the change to the existing order, such that the change order contains objects having attributes having any new values as indicated in the change to the existing order, while the existing order contains objects having attributes having values that are unchanged.

3. The method of claim 1 wherein the step of receiving a change to an existing order comprises the steps of:

receiving an identification of an existing order which is to be changed;

placing a hold on the existing order;

receiving a change signal indicating a new value for an attribute of an object; and

wherein the step of generating a change order based on the existing order comprises the step of:

for each object in the existing order for which the change signal indicates a new value for an attribute of that object, performing the steps of:

i) copying the object and any attributes and associated values in the existing order to an object in the change order having corresponding attributes and associated values; and

ii) assigning the new value as indicated in the change signal to a value of a corresponding attribute of the object in the change order.

4. The method of claim 3 wherein the step of comparing the change order to the existing order comprises the step of:

for each object in the existing order for which the change signal indicates a new value for an attribute of a corresponding object in the change order, generating a change order result that identifies:

i) the new value of the attribute of the corresponding object in the change order; and

ii) the existing value of the corresponding attribute of the object in the existing order.

5. The method of claim 4 wherein the step of comparing is done concurrently with the step of generating a change order, such that the step of generating a change order result in the step of comparing is performed during the processing of each object in the existing order for which the change signal indicates a new value for an attribute of that object.

6. The method of claim 4 wherein the step of comparing the change order to the existing order is done after the step of generating a change order.

7. The method of claim 4 wherein:

there are multiple objects in the existing order which correspond to the object in the change order having an attribute for which the change signal indicates a new value; and

wherein the step of comparing generates a change order result that indicates each existing value of the corresponding attribute of each of the multiple objects in the existing order.

8. The method of claim 1 wherein the step of comparing the change order to the existing order comprises the step of:

for each object having an attribute in the change order that has a different value from an existing value of a corresponding attribute of a corresponding object in the existing order, generating a change order result that identifies:

i) the value of the attribute of the object in change order; and

ii) the existing value of the corresponding attribute of the corresponding object in the existing order.

9. The method of claim 8 wherein the step of generating a change order result generates a change order result in a format including at least one of text and a markup language.

10. The method of claim 9 wherein the step of generating a change order result selects the format of the change order result based on an identity of a recipient of the change order result and wherein the step of providing provides the change order result to a recipient in the format selected based on the identity of the recipient.

11. A method for comparing order objects, the method comprising the steps of:

receiving a new value for an existing attribute of an existing peer object in an existing order;

copying the existing order to a change order such that the change order includes a peer object corresponding to the existing peer object, the peer object including a peer attribute corresponding to the existing attribute;

assigning the new value to the peer attribute of the peer object in the change order;

comparing the existing peer object in the existing order to the peer object in the change order to produce a change order result indicating differences between existing attribute and the peer attribute; and

providing the change order result to at least one recipient.

12. The method of claim 11 wherein the step of comparing invokes comparison logic which performs the steps of:

receiving an identity of the existing peer object in the existing order;

receiving an identity of the peer object in the change order; and

generating a change order result by comparing an existing value of the existing attribute of the existing peer object in the existing order to the new value of the peer attribute of the peer object in the change order to produce a change order result indicating the new value in comparison to the existing value.

13. The method of claim 12 wherein the step of comparing further comprises the steps of:

determining if any other attributes related to the peer object are changed based on the new value, and if the other attributes are changed, invoking the comparison logic on the peer objects related to those other attributes to produce a change order result indicating the differences between those other attributes.

14. In an order processing system, a method for comparing orders, the method comprising the steps of

receiving an identity of an existing order;

receiving an identity of a change order; and

generating a change order result by comparing an existing value of an existing attribute of an existing object in the existing order to a value of a peer attribute of a peer

object in the change order to produce a change order result indicating differences between the existing order and the change order.

15. An order processing computer comprising:

an interface;

a processor;

a memory encoded with an order application; and

an interconnection mechanism coupling the interface, the processor and the memory;

wherein the processor performs the order application as an order process to process changes to orders in an order processing system by performing the operations of:

receiving, via the interface, a change to an existing order, the existing order existing in an order database accessible to the order process;

generating, in the memory, a change order based on the existing order, the change order containing the change to the existing order;

comparing the change order to the existing order to generate a change order result in the memory that indicates differences between the change order and the existing order; and

providing the change order result to at least one recipient via the interface, such that the recipient may distinguish the differences between the change order and the existing order.

16. The order processing computer of claim 15 wherein when the order process performs the operation of generating a change order containing the change to the existing order, the order process performs the operations of:

copying the existing order to the change order in the memory such that the change order contains any objects that exist within the existing order, each object having at least one attribute and an associated value; and

replacing values of any attributes of objects in the change order in the memory with new values for those attributes as indicated in the change to the existing order, such

that the change order contains objects having attributes having any new values as indicated in the change to the existing order, while the existing order contains objects having attributes having values that are unchanged.

17. The order processing computer of claim 15 wherein when the order process performs the operation of receiving a change to an existing order, the order process performs the operations of:

receiving an identification of an existing order which is to be changed;

placing a hold on the existing order;

receiving a change signal via the interface indicating a new value for an attribute of an object; and

wherein when the order process performs the operation of generating a change order based on the existing order, the order process performs the operations of:

for each object in the existing order in the memory for which the change signal indicates a new value for an attribute of that object, performing the steps of:

i) copying the object and any attributes and associated values in the existing order to an object in the change order in the memory having corresponding attributes and associated values; and

ii) assigning the new value as indicated in the change signal to a value of a corresponding attribute of the object in the change order.

18. The order processing computer of claim 17 wherein when the order process performs the operation of comparing the change order to the existing order, the order process performs the operations of:

for each object in the existing order in the memory for which the change signal indicates a new value for an attribute of a corresponding object in the change order in the memory, generating a change order result in memory that identifies:

i) the new value of the attribute of the corresponding object in the change order; and

ii) the existing value of the corresponding attribute of the object in the existing order.

19. The order processing computer of claim 18 wherein the order process performs the operation of comparing concurrently with the operation of generating a change order, such that the operation of generating a change order result in the operation of comparing is performed during the processing of each object in the existing order for which the change signal indicates a new value for an attribute of that object.

20. The order processing computer of claim 18 wherein the order process performs the operation of comparing the change order to the existing order after the step of generating a change order.

21. The order processing computer of claim 18 wherein:

there are multiple objects in the existing order in the memory which correspond to the object in the change order having an attribute for which the change signal indicates a new value; and

wherein when the order process performs the operation of comparing, the order process generates a change order result in the memory that indicates each existing value of the corresponding attribute of each of the multiple objects in the existing order.

22. The order processing computer of claim 15 wherein when the order process performs the operation of comparing the change order to the existing order, the order process performs the operations of:

for each object having an attribute in the change order that has a different value from an existing value of a corresponding attribute of a corresponding object in the existing order in the memory, generating a change order result in the memory that identifies:

i) the value of the attribute of the object in change order; and

ii) the existing value of the corresponding attribute of the corresponding object in the existing order.

23. The order processing computer of claim 15 wherein when the order process performs the operation of generating a change order result, the order process generates a change order result in a format including at least one of text and a markup language.

24. The order processing computer of claim 23 wherein when the order process performs the operation of generating a change order result, the order process selects a format of the change order result based on an identity of a recipient of the change order result.

25. The order processing computer of claim 24 wherein when the order process performs the operation of providing, the order process provides the change order result from the memory to a recipient via the interface according to a format based on the identity of the recipient.

26. A processor in a computer system operating an order application to form an order process for comparing order objects in a computer system, the order process performing the operations of:

receiving, via an interface of the computer system, a new value for an existing attribute of an existing peer object in an existing order in an order database of the computer system;

copying the existing order to a change order in the order database, such that the change order includes a peer object corresponding to the existing peer object, the peer object including a peer attribute corresponding to the existing attribute;

assigning the new value to the peer attribute of the peer object in the change order in the order database;

comparing the existing peer object in the existing order to the peer object in the change order to produce a change order result in the computer system indicating differences between existing attribute and the peer attribute; and

providing the change order result from the computer system to at least one recipient.

27. The order process of claim 26 wherein when the order process performs the operation of comparing, the order process invokes comparison logic on a processor in the computer system, the comparison logic performing the operations of:

receiving an identity of the existing peer object in the existing order;

receiving an identity of the peer object in the change order; and

generating a change order result by comparing an existing value of the existing attribute of the existing peer object in the existing order to the new value of the peer attribute of the peer object in the change order to produce a change order result indicating the new value in comparison to the existing value.

28. The method of claim 26 wherein the step of comparing further comprises the steps of:

determining if any other attributes related to the peer object are changed based on the new value, and if the other attributes are changed, invoking the comparison logic on the peer objects related to those other attributes to produce a change order result indicating the differences between those other attributes.

29. A processor in a computer system performing comparator logic instructions to:

receive an identity of an existing order in the processor;

receive an identity of a change order in the processor;

generating a change order result in the processor by comparing an existing value in the processor of an existing attribute of an existing object in the existing order to a value in the processor of a peer attribute of a peer object in the change order to produce a change order result indicating differences between the existing order and the change order.

30. A computer program product having a computer-readable medium including order application computer program logic encoded thereon for processing changes to orders in an order processing system, such that the computer program logic, when performed on at least one processor within a computer system, causes the at least one processor to perform the operations of:

receiving a change to an existing order;

generating a change order based on the existing order, the change order containing the change to the existing order;

comparing the change order to the existing order to generate a change order result that indicates differences between the change order and the existing order; and

providing the change order result to at least one recipient such that the recipient may distinguish the differences between the change order and the existing order.

31. A computer program product having a computer-readable medium including order application computer program logic encoded thereon for processing changes to orders in an order processing system, such that the computer program logic, when performed on at least one processor within a computer system, causes the at least one processor to perform the operations of:

receiving a new value for an existing attribute of an existing peer object in an existing order;

copying the existing order to a change order such that the change order includes a peer object corresponding to the existing peer object, the peer object including a peer attribute corresponding to the existing attribute;

assigning the new value to the peer attribute of the peer object in the change order;

comparing the existing peer object in the existing order to the peer object in the change order to produce a change order result indicating differences between existing attribute and the peer attribute; and

providing the change order result to at least one recipient.

32. A computer program product having a computer-readable medium including comparator logic encoded thereon for generating a change order result in an order processing system, such that the comparator logic, when performed on at least one processor within a computer system, causes the at least one processor to perform the operations of:

receiving an identity of an existing order;

receiving an identity of a change order; and

generating a change order result by comparing an existing value of an existing attribute of an existing object in the existing order to a value of a peer attribute of a peer object in the change order to produce a change order result indicating differences between the existing order and the change order.

33. An order processing computer comprising:

an interface;

a processor;

a memory encoded with an order application; and

an interconnection mechanism coupling the interface, the processor and the memory;

wherein the processor performs the order application as an order process to process changes to orders in an order processing system such that the order processing computer includes:

means for receiving a change to an existing order, the existing order existing in an order database accessible to the order process;

means for generating, in the memory, a change order based on the existing order, the change order containing the change to the existing order;

means for comparing the change order to the existing order to generate a change order result in the memory that indicates differences between the change order and the existing order; and

means for providing the change order result to at least one recipient via the interface, such that the recipient may distinguish the differences between the change order and the existing order.

34. A processor in a computer system operating an order application to form an order process for comparing order objects in a computer system, the order process providing:

means for receiving, via an interface of the computer system, a new value for an existing attribute of an existing peer object in an existing order in an order database of the computer system;

means for copying the existing order to a change order in the order database, such that the change order includes a peer object corresponding to the existing peer object, the peer object including a peer attribute corresponding to the existing attribute;

means for assigning the new value to the peer attribute of the peer object in the change order in the order database;

means for comparing the existing peer object in the existing order to the peer object in the change order to produce a change order result in the computer system indicating differences between existing attribute and the peer attribute; and

means for providing the change order result from the computer system to at least one recipient.